

**TÍTULO:** Práctica Programación Multihilo: La Cena de los Filósofos **ALUMNO:** Jose Antonio Serrano Santos **FECHA:** 12/12/2025

**1. Introducción** En esta práctica se ha desarrollado una solución concurrente al problema de la Cena de los Filósofos utilizando Java. El objetivo principal ha sido coordinar los hilos para evitar condiciones de carrera y el interbloqueo (deadlock) mediante el uso de un Monitor.

**2. Diseño de la Solución** Se ha implementado el patrón Monitor, donde una clase central ([MonitorFilosofos](#)) gestiona el acceso a los recursos compartidos (tenedores) mediante métodos sincronizados.

- **Clase Filosofo:** Implementa [Runnable](#). Ciclo infinito de Pensar -> Comer.
- **Clase Monitor:** Controla el array de estados y usa [wait\(\) / notifyAll\(\)](#).

**3. Pruebas y Depuración (Debug)** Se ha utilizado el depurador de Eclipse para verificar que los hilos se bloquean correctamente cuando los recursos no están disponibles.

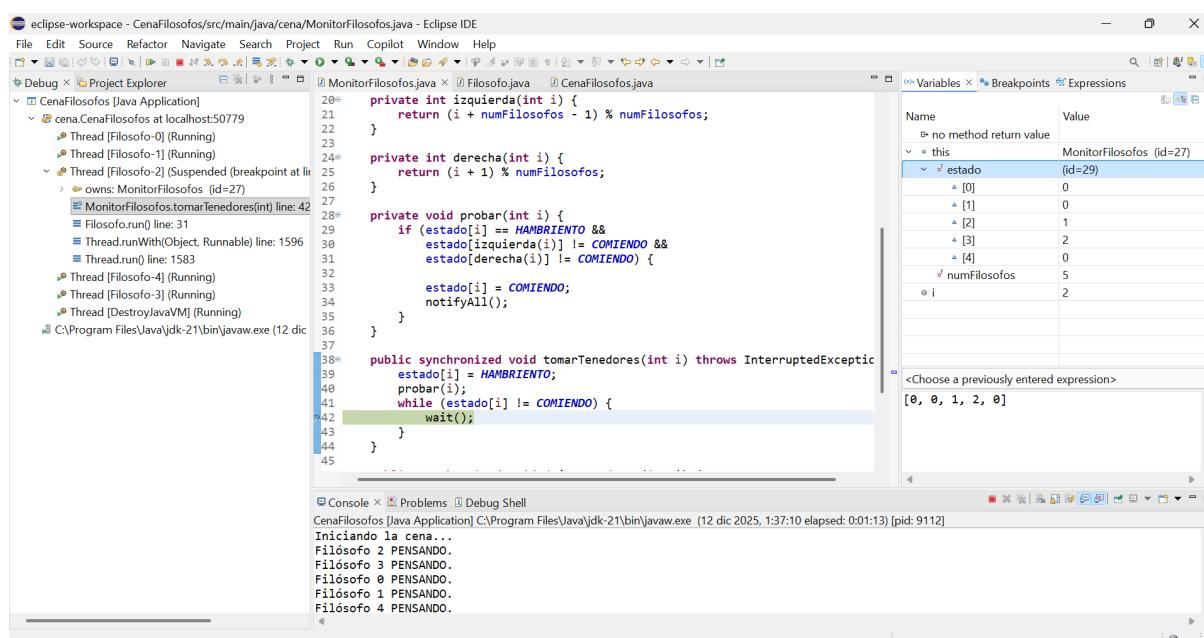


Fig 1. Captura del depurador mostrando un hilo en estado WAITING esperando por los tenedores.

**4. Pruebas Unitarias (JUnit)** Se han realizado pruebas automáticas para validar la lógica de transición de estados y asegurar que un filósofo libera correctamente los recursos.

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - CenaFilosofos/src/test/java/cena/MonitorTest.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Copilot Window Help
- Toolbar:** Standard Eclipse toolbar with icons for file operations.
- Left Sidebar:** Package Explorer, JUnit (highlighted), and a Failure Trace section.
- Central Area:** Monitored code editor showing `MonitorTest.java`. The code is a JUnit test for a monitor class. It includes imports for `org.junit.Assert.*` and `org.junit.Test`, and defines a test method `testCicloVida` that creates a monitor for 5 philosophers, simulates them taking and releasing forks, and checks their states (PENSANDO, COMIENDO).
- Status Bar:** Finished after 0,061 seconds, Runs: 1/1, Errors: 0, Failures: 0.

Fig 2. Resultado exitoso (barra verde) de los tests unitarios.

**5. Conclusión** La práctica funciona correctamente. Se ha verificado mediante JUnit que la lógica del monitor es sólida y mediante el depurador que la gestión de hilos es eficiente, cumpliendo con los requisitos de sincronización.